

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In the Application of:	)	
	)	
James P. Gustafson	)	
	)	
Serial No. 10/790,340	)	
	)	
Filed: March 1, 2004	)	Electronically Filed on
	)	
For: Mobile Handset With Efficient	)	October 5, 2009
Interruption Point Detection During A	)	
Multiple-Pass Update Process	)	
	)	
Examiner: Wei, Zheng	)	
	)	
Group Art Unit: 2192	)	
	)	
Confirmation No. 9396	)	
	)	

APPEAL BRIEF

Mail Stop Appeal Brief – Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Dear Sir:

Applicant respectfully requests that the Board of Patent Appeals and Interferences reverse the final rejection of claims 1-24 of the present application. This Appeal Brief is timely because it is being filed with a Notice of Appeal within three months of the mailing of the Final Office action mailed July 6, 2009.

**REAL PARTY IN INTEREST**  
**(37 C.F.R. § 41.37(c)(1)(ii))**

The real party in interest is Hewlett-Packard Development Company, L.P., having a place of business in Houston, Texas.

**RELATED APPEALS AND INTERFERENCES**  
**(37 C.F.R. § 41.37(c)(1)(iii))**

Not applicable.

**STATUS OF THE CLAIMS**  
**(37 C.F.R. § 41.37(c)(1)(iii))**

The present application includes claims 1-24, all of which remain rejected.<sup>1</sup> Applicant identifies claims 1-24 as the claims that are being appealed. The text of the pending claims is provided in the Claims Appendix.

**STATUS OF AMENDMENTS**  
**(37 C.F.R. § 41.37(c)(1)(iv))**

Subsequent to the final rejection of claims 1-24 mailed July 06, 2009, Applicant filed a Notice of Appeal on October 5, 2009. Applicant did not amend any of the claims after the final Office Action was mailed.

---

<sup>1</sup> See July 06, 2009 Final Office Action.

**SUMMARY OF CLAIMED SUBJECT MATTER**  
**(37 C.F.R. § 41.37(c)(1)(v))**

**Independent claim 1:**

Independent claim 1 recites a mobile handset.<sup>2</sup> The mobile handset comprises at least one memory device that stores at least one of a firmware component and a software component.<sup>3</sup> The mobile handset also comprises an update agent capable of updating the at least one of a firmware component and a software component employing an update process that comprises a plurality of transform passes.<sup>4</sup> Each transform pass is associated with its own memory bank order, and each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass.<sup>5</sup> Also, the update agent executes at least one of the plurality of transform passes in a fault-tolerant mode.<sup>6</sup>

**Independent claim 11 recites the following:**

Independent claim 11 recites a mobile handset comprising a plurality of memory banks containing at least one of a firmware and a plurality of software components.<sup>7</sup> The mobile handset comprises an update package comprising a difference information.<sup>8</sup> The mobile handset also comprises an update agent capable of updating at least a

---

<sup>2</sup> See present application at, e.g., page 3, line 2, page 8, lines 2-3, Figure 1, ref. 107.

<sup>3</sup> See *id.*, e.g., at page 3, lines 2-3, page 8, lines 9-13; Figure 1, refs. 111, 115, 117, 121, 125.

<sup>4</sup> See *id.*, e.g., at page 3, lines 3-6, page 8, lines 6-9, Figure 1, ref. 113.

<sup>5</sup> See *id.*, e.g., at page 4, lines 12-15, page 9, lines 25-26, page 10, lines 22-30.

<sup>6</sup> See *id.*, e.g., at page 3, lines 6-9, page 10, lines 24-25 and 29-30.

<sup>7</sup> See *id.*, e.g., at page 4, lines 7-9, page 8, lines 2-6 and 17-21, Figure 1, refs. 107, 111, 115, 117, 121.

portion of the at least one of firmware and a plurality of software components,<sup>9</sup> the update agent employing an update process that comprises a plurality of transform passes.<sup>10</sup> Each of the plurality of transform passes is associated with its own memory bank order, and each transform pass updates the plurality of memory banks in the associated memory bank order for that transform pass.<sup>11</sup> The mobile handset further comprises a set of decision maker banks identified within the plurality of memory banks,<sup>12</sup> one decision maker bank for each of the plurality of transform passes.<sup>13</sup> The set of decision maker banks is used to determine which of the plurality of transform passes was interrupted,<sup>14</sup> in order that the update process may be subsequently reattempted beginning with the interrupted transform pass.<sup>15</sup>

**Independent claim 19:**

Independent claim 19 recites a method for recovering from interruption of a fault-tolerant process of updating a mobile handset comprising a plurality of memory banks from a first firmware version to a second firmware version, the update process comprising a plurality of transform passes and having a transform pass order, each of the plurality of transform passes performing a transform upon the plurality of memory

---

<sup>8</sup> See *id.*, e.g., at page 4, lines 10-11, page 8, lines 23-26.

<sup>9</sup> See *id.*, e.g., at page 4 lines 11-12, page 8, lines 6-9, Figure 1, ref. 113.

<sup>10</sup> See *id.*, e.g., at page 4, lines 12-13, page 8, lines 6-9.

<sup>11</sup> See *id.*, e.g., at page 4, lines 13-15, page 9, lines 25-26, page 10, lines 22-30.

<sup>12</sup> See *id.*, e.g., at page 4, lines 15-16, page 10, lines 18-21.

<sup>13</sup> See *id.*, e.g., at page 4, lines 16-17, page 10, lines 18-21.

<sup>14</sup> See *id.*, e.g., at page 4, lines 17-18, page 10, lines 11-15 and 18-30.

<sup>15</sup> See *id.*, e.g., at page 4, lines 18-20, page 10, lines 11-15 and 18-30.

banks in a memory bank order.<sup>16</sup> The method comprises determining as a recovery transform pass, one of the plurality of transform passes interrupted during the update process,<sup>17</sup> wherein each of the plurality of transform passes is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass.<sup>18</sup> The method also comprises determining as a recovery memory bank, one of the plurality of memory banks in the associated memory bank order for the interrupted transform pass during which update processing was interrupted.<sup>19</sup> Additionally, the method comprises invoking the update process by performing an update of the recovery memory bank using the recovery transform pass.<sup>20</sup>

**GROUND OF REJECTION TO BE REVIEWED ON APPEAL  
(37 C.F.R. § 41.37(c)(1)(vi))**

- Claims 1-24 stand rejected under 35 U.S.C. § 102(e) as being anticipated by United States Patent No. 7,082,549 ("Rao").

**ARGUMENT  
(37 C.F.R. § 41.37(c)(1)(vii))**

The Examiner has maintained the rejections of claims 1-24. As shown above,

---

<sup>16</sup> See *id.*, e.g., at page 5, lines 13-18, page 11, lines 8-12, Figure 2.

<sup>17</sup> See *id.*, e.g., at page 5, lines 19-21, page 12, lines 3-9, Figure 2, ref. 215.

<sup>18</sup> See *id.*, e.g., at page 4, lines 13-15, page 9, lines 25-26, page 10, lines 22-30.

<sup>19</sup> See *id.*, e.g., at page 5, lines 21-23, page 12, lines 3-9, Figure 2, ref. 215.

<sup>20</sup> See *id.*, e.g., at page 5, lines 23-25, page 12, lines 11-12; Figure 2, ref. 221.

Rao forms the basis for all the claim rejections.

Claims 1-10 should be in condition for allowance at least because Rao does not teach or disclose "an update agent capable of updating the at least one of a firmware component and a software component employing an update process that comprises a plurality of transform passes, wherein each transform pass is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass," as recited in claim 1.

Claims 11-18 should be in condition for allowance at least because Rao does not teach or disclose "an update agent capable of updating at least a portion of the at least one of firmware and a plurality of software components, the update agent employing an update process that comprises a plurality of transform passes, wherein each of the plurality of transform passes is associated with its own memory bank order, and wherein each transform pass updates the plurality of memory banks in the associated memory bank order for that transform pass," or "a set of decision maker banks identified within the plurality of memory banks, one decision maker bank for each of the plurality of transform passes, the set of decision maker banks used to determine which of the plurality of transform passes was interrupted, in order that the update process may be subsequently reattempted beginning with the interrupted transform pass," as recited in claim 11.

Claims 19-24 should be in condition for allowance at least because Rao does not

teach or disclose “wherein each of the plurality of transform passes is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass,” as recited in claim 19.

#### I. **Rao Does Not Anticipate Claims 1-24**

Before addressing the specific rejections at issue, Applicant begins by addressing certain requirements for a finding of anticipation. “A claim is anticipated only if **each and every element** as set forth in the claim is found, either expressly or inherently described, in **a single prior art reference.**” *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987) (emphasis added). “The **identical** invention must be shown in as complete detail as is contained in ... the claim.” *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989) (emphasis added).

Turning first to claim 1 and the claims that depend therefrom, independent claim 1 recites “[a] mobile handset comprising: at least one memory device that stores at least one of a firmware component and a software component; an update agent capable of updating the at least one of a firmware component and a software component employing an update process that comprises a plurality of transform passes, wherein each transform pass is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank

order for that transform pass; and the update agent executing at least one of the plurality of transform passes in a fault-tolerant mode.” As detailed below, Applicant respectfully submits that Rao fails to describe, teach or suggest at least “a plurality of transform passes, wherein each transform pass is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass.” Indeed, a previous Office Action expressly recognized that Rao “does not explicitly disclose each transform pass is associated with its own memory bank order.” (See June 12, 2008 Office Action at p. 3.) Only after Applicant demonstrated the inappropriateness of that Office Action’s obviousness rejections did subsequent Office Actions revert to asserting Rao as an anticipatory reference.

In any event, claim 1 expressly recites that “each transform pass is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass.” Thus, as explicitly required by claim 1, each transform pass must update a plurality of memory banks in an associated memory bank order. The purported “transform passes” of Rao asserted by the Office Action do not teach a plurality of transform passes that each update a plurality of memory banks in an associated memory bank order. As such, Rao does not anticipate claim 1.

For example, in rejecting claim 1, the Office Action asserts that Rao discloses

an update agent capable of updating the at least one of a  
firmware component and a software component employing  
an update process that comprises a plurality of transform



passes, wherein each transform pass is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass (see for example, Fig. 2, step 207-223 and related text; also see, col. 2, lines 62-65, "...selecting one of the plurality of banks using, ...a specified bank order" and col. 5 line 61 – col. 6, line[] 9, "The bank order specification may be used by update agent 117 to direct the sequence of the update of the various banks of memory such as bank 1 119 through bank N 131, as necessary, in the non-volatile memory 111"; also see Fig.3, step 311-321 and related text)

(Office Action at p. 5.) The Office Action also cites "Fig. 1, item 117, 'Update Agent' and related text" in rejecting a generally similar limitation in claim 11. (See *id.* at p. 10.)

As also indicated in previous submissions (see, e.g., Response filed March 16, 2009), Applicant respectfully submits these assertions, as explained by the Office Action, cannot properly support an anticipation rejection. As an initial matter, it appears that the Office Action attempts, at various points, to assert either the entire update process of all blocks as a purported "transform pass," and, at other points, to assert the update of a single block as a purported "transform pass." Regardless of which of these aspects of Rao the Office Action attempts to rely upon, neither of those aspects of Rao (or any other aspect of the cited and discussed portions of Rao in the Office Action) discloses the plurality of transform passes as claimed.

To the extent the Office Action relies upon the entire update process of Rao as disclosing a transform pass that specifies a "specified bank order," the entire update process cannot disclose a **plurality** of transform passes, but instead could disclose, at

most, one pass. Nor does such a single sequence "such as bank 1 119 through bank N 131" teach wherein each of a plurality of transform passes is associated with its own memory bank order. On the other hand, to the extent the Office Action relies upon the update (or portion of an update) of a single memory bank in Rao as a "transform pass," the update of a single bank, by definition, cannot update a **plurality of memory banks in the associated memory bank order** for that transform pass, as required by claim 1 for each transform pass. Such an update of a **single** memory bank cannot disclose a transform pass updating a **plurality** of memory banks, let alone a transform pass updating a **plurality of memory banks in an associated memory bank order** as claimed. Thus, neither the entire update process of Rao (at least because it does not disclose a plurality of transform passes, or wherein each of the plurality of transform passes is associated with its own memory bank order) nor the update of a single memory bank (at least because it does not disclose updating a plurality of memory banks in the associated memory bank order as claimed) can be relied upon to teach the plurality of transform passes as fully set forth by claim 1.

Turning now to specifically address the cited portions of Rao, Applicant begins with the first cited portion (namely step 207-223 and related text). A portion of Rao describing those steps reads as follows:

FIG. 2 is a flow chart illustrating an exemplary method of startup of a mobile handset such as the mobile handset 107 of FIG. 1, in which a fault tolerant update agent is invoked to conduct firmware/software update operations, in accordance with the present invention. The processing

starts when the mobile handset is powered up or rebooted (block 207). Next, the initialization or boot sequence is executed (block 209). A determination is then made whether an update of the firmware/software of the mobile handset is to be conducted (block 211). For example, in one embodiment, a status table may be accessed to determine if a flag in the status table indicates a need to update firmware/software using an update package previously downloaded and available in the mobile handset.

If it is determined that an update is not necessary, then the regular startup procedure of the mobile handset is executed (block 225) and startup processing terminates (block 221). If, however, it is determined that an update is necessary, then the point where the fault-tolerant update agent should resume the update process is determined (block 213). For example, that point may be at the beginning, for a newly begun update operation, or at a specific bank of non-volatile memory, in the case of a previously initiated but interrupted update operation. Next, the various operations related to the fault-tolerant update process are conducted by the update agent (block 215). finally, the update agent initiates a reboot of the mobile handset (block 223).

(Rao, 8:14-40.) Thus, this portion of Rao is silent with respect to any mention of memory bank order, let alone to “a plurality of transform passes, wherein each of the plurality of transform passes is associated with its own memory bank order...” as claimed. For instance, even assuming, *arguendo*, that an “update operation” could be interpreted as a single transform pass, this portion would still be silent with respect to a **plurality** of transform passes as claimed. Applicants recognize that this portion of Rao mentions a “fault-tolerant update process.” However, such a teaching cannot be stretched so far as to somehow anticipate a plurality of transform passes as expressly set forth in claim 1.

The next cited portion, column 2, lines 62-65, merely states as follows: "...update instruction and selecting one of the plurality of banks using at least one of at least a cyclic redundancy check, a message digest, a digital signature, a checksum, and a specified bank order. Such a method may also comprise...." Applicants note that this cited portion references a "specified bank order" and "selecting one of the plurality of banks." Such a mere mention of a specified bank order or the selection of one of a plurality of banks, however, teaches nothing with respect to a **plurality of transform passes** as claimed, or wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass, let alone the plurality of transform passes as fully set forth by claim 1. Again, if the "specified bank order" and/or "selecting one of the plurality of banks" relates to the entire update, it cannot disclose "a plurality of transform passes" as fully set forth by claim 1. If, on the other hand, the Office Action asserts it as relating to a single memory bank being updated in a bank-by-bank process, such a purported "transform pass" does not disclose "wherein each transform pass updates a plurality of memory banks..."

The next relied upon portion of Rao is col. 5, line 61 – col. 6, line 9. That portion reads as follows:

The update agent 117 in an embodiment of the present invention may employ a bank order specification. The bank order specification may be determined during the creation of the update package, and may be received from the update package repository 137 as part of the corresponding update package. The bank order specification may be used by update agent 117 to direct the

sequence of the update of the various banks of memory such as bank 1 119 through bank N 131, as necessary, in the non-volatile memory 111. The use of a bank order specification may permit the size of the update package to be optimized by controlling the bank update sequence. The update agent 117 may copy each of the bank 1 119 through bank N 131 to the working bank 125 according to the bank order specification, so that they may be modified or updated before they are written back into non-volatile memory area 111.

Applicant respectfully submits that such a "bank order specification" as described in the above cited portion of Rao relates to a **single** sequence. The cited portion begins, for example, "The update agent 117 in an embodiment of the present invention may employ a bank order specification." (emphasis added.) Moreover, the bank order specification is used to update banks of memory "such as bank 1 119 through bank N 131, as necessary." Thus, this portion is silent with respect to a plurality of transform passes, let alone with respect to wherein each of the plurality of transform passes is associated with its own memory bank order, as it discusses only one bank order specification. Thus, even if somehow assumed, *arguendo*, as teaching other aspects of claim 1, this portion still could possibly teach, at most, one "transform pass" (to the extent the Office Action asserts such an update of all of memory banks as a transform pass).

In connection with its rejection of generally similar aspects of claim 11, the Office Action also references "Fig. 1, item 117, 'Update Agent' and related text," which would appear to include col. 5, line 61 – col. 6, line 9, cited above. Applicant has reviewed the remainder of the discussion of Fig. 1. That discussion spans several columns and thus

is not repeated in its entirety herein. Applicant does not understand any portion of that discussion to remedy the failure of the above discussed portions to disclose the presently claimed subject matter at issue. Applicant has previously requested a more specific explanation of how any of that portion of Rao's disclosure teaches the presently claimed subject matter, to the extent it was intended to assert any part of that discussion as remedying the above discussed shortcomings. (See, e.g., Response filed March 16, 2009 at p. 14.) Subsequent to that request, the Office Action did not provide any identification or explanation. Applicants thus understand the above discussion to fully address any assertions regarding "Fig. 1, item 117, 'Update Agent' and related text."

The Office Action additionally cites "Fig. 3, step 311-321 and related text." The Office Action further relies on that portion in responding to Applicant's previous discussion of shortcomings in the disclosure of Rao. In responding to that discussion, the Office Action states as follows:

At page 8, last paragraph, the Applicants submit that the amended Claim 1 requires a plurality of transform passes, wherein each transform pass is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass. Prior art Rao cannot anticipate claim 1 at least because Rao does not teach such a plurality of transform passes with each transform pass performed in a memory bank order associated with that particular transform pass.

However, Examiner's position is that Rao still teaches such limitations as amended. It should be noted that "transform pass" as recited in the claim can be reasonable

interpreted as an update step in bank by bank manner as in Rao (see for example, col.2, line 66-col.3, line 3, "converting the selected bank from the first code version to the second code version using that at least one update instruction and repeating the receiving, selecting and converting in a bank by bank manner until each of the plurality of banks has been updated") and "a plurality of memory banks" can also be read as all memory banks including "original bank", "working bank" and "backup bank" which are involved in each transform pass updating circle in the specific order. (see Rao for example, Fig.3, steps 311->313->315->317->321->311 and related text). Therefore, Rao's each bank by bank update step (transform pass) to update plurality of banks including original bank, working bank and backup bank teaches the limitation as recited in claim 1. Moreover, for each bank selecting for update (transform pass), it is according to (associated) the specified bank order in update package (memory bank order) (see for example, col.2, line 41-42, "selecting may use a specified bank order at all other times").

(Office Action at p. 2-3.) Applicants respectfully traverse these assertions, and further respectfully submit that this explanation offered by the Office Action does nothing to remedy the above discussed shortcomings in the teachings of Rao with respect to the presently claimed subject matter.

For example, the Office Action asserts that "'transform pass' as recited in the claim can be reasonable [sic] interpreted as an update step in bank by bank manner as in Rao (see for example, col.2, line 66-col.3, line 3, 'converting the selected bank from the first code version to the second code version using that at least one update instruction and repeating the receiving, selecting and converting in a bank by bank manner until each of the plurality of banks has been updated')." Again, a mere

disclosure of a bank by bank process does not disclose the presently claimed subject matter, which expressly requires, *inter alia*, a plurality of transform passes, each of the transform passes updating a plurality of memory banks in an associated bank order. The updating of just one bank cannot disclose a transform pass as claimed, because a claimed transform pass must update a plurality of banks, not just one bank. And the update of all of the banks cannot disclose a plurality of transform passes as required by claim 1 (for example, a plurality of transform passes with each of the transform passes having an associated memory bank order), but instead would teach, at most, just one transform pass -- with just one order for all the banks, there can be, at most, just one "transform pass."

The Office Action also asserts that "a plurality of memory banks' can also be read as all memory banks including 'original bank', 'working bank' and 'backup bank' which are involved in each transform pass updating circle in the specific order. (see Rao for example, Fig.3, steps 311->313->315->317->321->311 and related text)." However, claim 1 expressly requires, *inter alia*, that each transform pass updates a plurality of memory banks. In contrast, the purported "plurality of memory banks" asserted by the Office Action are not all updated, as expressly required by claim 1. Referring to Fig. 3 of Rao, step 311 is "Copy Original Bank to Working Bank." Step 313 is "Update Working Bank." Step 315 is "Copy Working Bank to Backup Bank." Step 317 is "Copy Working Bank to Original Bank," and step 321 is "Advance to Next Bank." Thus, one "cycle" through these steps cannot teach the update of a plurality of banks,



as only one bank, the working bank, is updated. Whether other banks are somehow "involved" is irrelevant. Claim 1 explicitly requires that, for a given transform pass, a plurality of memory banks are **updated**. Therefore, one cycle through these steps cannot teach a transform pass as claimed. Again, going through this cycle multiple times to perform an update does not teach a plurality of transform passes, each having an associated memory bank order. For example, a mere teaching of one "specified bank order" for an entire update does not teach a plurality of transform passes, each with their own associated memory bank order.

Accordingly, for at least the reasons discussed above, as well as in previous responses, Applicants respectfully submit that the cited portions discussing a bank-by-bank update in Rao do not disclose the subject matter claimed by claim 1. Applicant therefore respectfully submits that Rao does not anticipate claim 1 or any of its dependent claims, and that those claims are allowable over Rao.

Applicant now turns to independent claim 11 and its dependent claims. Independent claim 11 recites "[a] mobile handset comprising a plurality of memory banks containing at least one of a firmware and a plurality of software components, the mobile handset comprising: an update package comprising a difference information; an update agent capable of updating at least a portion of the at least one of firmware and a plurality of software components, the update agent employing an update process that comprises a plurality of transform passes, **wherein each of the plurality of transform passes is associated with its own memory bank order, and wherein each**

**transform pass updates the plurality of memory banks in the associated memory bank order for that transform pass**; and a set of decision maker banks identified within the plurality of memory banks, one decision maker bank for each of the plurality of transform passes, the set of decision maker banks used to determine which of the plurality of transform passes was interrupted, in order that the update process may be subsequently reattempted beginning with the interrupted transform pass." Thus, generally similar to claim 1, claim 11 requires a plurality of transform passes, with each transform pass updating a plurality of memory banks in a memory bank order associated with that transform pass. For at least the reasons discussed above in connection with claim 1, Rao does not teach at least those aspects of claim 11.

Further, Claim 11 also recites "a set of decision maker banks identified within the plurality of memory banks, one decision maker bank for each of the plurality of transform passes, the set of decision maker banks used to determine which of the plurality of transform passes was interrupted, in order that the update process may be subsequently reattempted beginning with the interrupted transform pass." Thus, claim 11 relates to, *inter alia*, a set of decision maker banks, with one decision maker bank associated each of the plurality of transform passes.

The Office Action cites col. 8, lines 28-62 and col. 5, lines 61 – col. 6, line 9 in asserting that Rao discloses this aspect of claim 11.

The second cited portion (col. 5, lines 61 – col. 6, line 9) has been reproduced and discussed above. Column 8, lines 28-62 of Rao read as follows:

If it is determined that an update is not necessary, then the regular startup procedure of the mobile handset is executed (block 225) and startup processing terminates (block 221). If, however, it is determined that an update is necessary, then the point where the fault-tolerant update agent should resume the update process is determined (block 213). For example, that point may be at the beginning, for a newly begun update operation, or at a specific bank of non-volatile memory, in the case of a previously initiated but interrupted update operation. Next, the various operations related to the fault-tolerant update process are conducted by the update agent (block 215). Finally, the update agent initiates a reboot of the mobile handset (block 223).

FIG. 3 is a flow chart illustrating an exemplary method of performing a fault tolerant update of a mobile handset such as the mobile handset 107 of FIG. 1 wherein a working bank, a backup bank, and a plurality of updateable original banks are employed in the update process, in accordance with the present invention. The process starts when the mobile handset detects that an update of firmware/software is necessary, and the point in memory at which the update needs to begin or resume is determined (block 307). For example, in one embodiment of the present invention, the point at which update processing should begin or resume may be determined using a verification process in which a CRC value is computed for each bank of non-volatile memory being updated. The computed value for each bank is then compared to the corresponding pre-computed value in a list of CRC values provided within the update package containing the firmware/software. The comparison continues in a bank-by-bank fashion through the list until a bank of non-volatile memory is encountered where the computed CRC value fails to match the value in the list for that bank. The failed match indicates an anomaly or corruption, thus indicating the point at which the update activity should start or resume.

This portion of Rao relates to a "fault tolerant update of a mobile handset."

Similar to above, Applicants respectfully submit this portion of Rao is silent with respect to "a plurality of transform passes..." as claimed, and does not remedy the previously

discussed shortcomings of the other cited portions. In any event, Applicant further respectfully submits such a disclosure does not teach "a set of decision maker banks identified within the plurality of memory banks, one decision maker bank for each of the plurality of transform passes..." as claimed. For example, a mere disclosure that a bank is encountered where a CRC value fails to match does not disclose associating a decision maker bank with a particular transform pass, or a plurality of transform passes. This is even more so in light of the cited portions of Rao not disclosing the plurality of transform passes as claimed in the first place, as discussed above.

In responding to Applicant's previous discussion regarding claim 11, the Office Action states as follows:

At page 16, first paragraph, the Applicants submit that a mere disclosure that a bank is encountered where a CRC value fails to match does not disclose a decision maker bank for each of the plurality of transform passes, as such a teaching does not teach associating a decision maker bank with a particular transform pass, or a plurality of transform passes, let alone a decision maker bank as claimed.

However, Examiner respectfully disagrees. As Rao disclosed, the point or bank of resuming updating is determined by a verification process to compute value for each bank is then to compare to the corresponding pre-computed value in a list of CRC values provided within the update package (see for example, col.8, lines 28-62). Therefore, each memory bank of non-volatile memory is considered as a decision maker bank during updating process to determine if the previous bank update (transform pass) was interrupted (see for example, col. 8, lines 28-62, "...the failed match indicates an anomaly or corruption, thus indicating the point at which the update activity should start or resume").

(Office Action at p. 4.) Such an assertion does not properly support a finding of anticipation. For example, the Office Action relies upon a purported “decision maker bank” as being a memory bank associated with a single memory bank. The Office Action further characterizes a transform pass as a “previous bank update.” Such assertions directly contradict the express requirements of claim 11, which explicitly requires that a claimed transform pass updates a plurality of memory banks. Thus, the update of a single bank (i.e. the “previous bank update”) cannot disclose a transform pass (“wherein each transform pass updates the plurality of memory banks...”) as claimed. Similarly, a purported “decision maker” for a single bank (i.e. the previous bank) cannot be for a transform pass, as a transform pass as claimed updates a plurality of memory banks.

For at least the above reasons, as well as those discussed in previous submissions, Applicant respectfully submits that the cited portions of Rao do not disclose the subject matter claimed by claim 11, that Rao does not anticipate claim 11 or any of its dependent claims, and that those claims are allowable.

With respect to independent claim 19 and its dependent claims, Applicant begins by noting that claim 19 recites “A method for recovering from interruption of a fault-tolerant process of updating a mobile handset comprising a plurality of memory banks from a first firmware version to a second firmware version, the update process comprising a plurality of transform passes and having a transform pass order, each of the plurality of transform passes performing a transform upon the plurality of memory

banks in a memory bank order, the method comprising: determining as a recovery transform pass, one of the plurality of transform passes interrupted during the update process, wherein each of the plurality of transform passes is associated with its own memory bank order, and **wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass**; determining as a recovery memory bank, one of the plurality of memory banks in the associated memory bank order for the interrupted transform pass during which update processing was interrupted; and invoking the update process by performing an update of the recovery memory bank using the recovery transform pass.” Thus, for at least similar reasons as discussed above, Applicant respectfully submits that Rao does not anticipate claim 19 or any claim dependent therefrom, and further that those claims are allowable.

## II. Conclusion

For at least the reasons discussed above, Applicant respectfully submits that the pending claims are allowable in all respects. Therefore, the Board is respectfully requested to reverse the rejections of pending claims 1-24.

Date: October 5, 2009

Respectfully submitted,

Hewlett-Packard Company  
Intellectual Property Administration  
Legal Department, M/S 35  
P.O. Box 272400  
Fort Collins, CO 80527-2400

/Kevin E. Borg/  
Kevin E. Borg  
Reg. No. 51,486

**CLAIMS APPENDIX**  
**(37 C.F.R. § 41.37(c)(1)(viii))**

1. A mobile handset comprising:
  - at least one memory device that stores at least one of a firmware component and a software component;
  - an update agent capable of updating the at least one of a firmware component and a software component employing an update process that comprises a plurality of transform passes, wherein each transform pass is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass; and
  - the update agent executing at least one of the plurality of transform passes in a fault-tolerant mode.
2. The mobile handset according to claim 1 wherein the update agent executes each of the plurality of transform passes in a fault-tolerant mode.
3. The mobile handset according to claim 1 wherein the update process comprises a pre-processing pass and an update pass.
4. The mobile handset according to claim 1 wherein the update agent is capable of determining a point of interruption of the update process, so as to restart the update process from the point of interruption on a subsequent invocation of the update process.
5. The mobile handset according to claim 4 wherein the update agent is capable of determining whether a previous invocation of the update process was interrupted during a pre-processing pass or during an update pass.
6. The mobile handset according to claim 1 wherein the update agent is

capable of determining which of the plurality of transform passes was interrupted during a previous execution of the update process.

7. The mobile handset according to claim 1 wherein the mobile handset comprises:

- a plurality of memory banks;
- a set of special signatures comprising at least one special signature corresponding to each of the plurality of transform passes;
  - the at least one special signature being associated with the last of the memory banks that is updated in the corresponding transform pass;
  - at least a subset of the plurality of memory banks being modified in each of the plurality of transform passes; and
  - the update agent capable of determining which of the plurality of transform passes was interrupted during a previous execution of the update process.

8. The mobile handset according to claim 7 wherein the update agent determines the transform pass that was interrupted during a previous execution of an update process, and the specific one of the plurality of memory banks that was last to be successfully updated, in order to resume the update process.

9. The mobile handset according to claim 8 wherein the update agent compares, in a transform pass order, each special signature from the set of special signatures to a signature computed for the last bank to be updated in the corresponding transform pass, until a mismatch is detected, the mismatch indicating a point of interruption during a previous update attempt.

10. The mobile handset of claim 7 wherein the special signature comprises one of a cyclic redundancy check (CRC) value and an MD5 hash value.



11. A mobile handset comprising a plurality of memory banks containing at least one of a firmware and a plurality of software components, the mobile handset comprising:

an update package comprising a difference information;

an update agent capable of updating at least a portion of the at least one of firmware and a plurality of software components, the update agent employing an update process that comprises a plurality of transform passes, wherein each of the plurality of transform passes is associated with its own memory bank order, and wherein each transform pass updates the plurality of memory banks in the associated memory bank order for that transform pass; and

a set of decision maker banks identified within the plurality of memory banks, one decision maker bank for each of the plurality of transform passes, the set of decision maker banks used to determine which of the plurality of transform passes was interrupted, in order that the update process may be subsequently reattempted beginning with the interrupted transform pass.

12. The mobile handset of claim 11 wherein the update package comprises information identifying the set of decision maker banks.

13. The mobile handset of claim 11 wherein the difference information comprises a set of executable instructions for converting a first version of the at least a portion of the at least one of firmware and a plurality of software components, to a second version of the at least a portion of the at least one of firmware and a plurality of software components.

14. The mobile handset of claim 11 wherein the update agent is capable of:

i) computing a signature for a decision maker bank in the set of decision

maker banks, beginning with the decision maker bank for the first transform pass in the transform pass order;

ii) comparing the computed signature to a corresponding predetermined signature for that transform pass, the predetermined signature contained in the update package, to determine whether a match exists;

iii) repeating (i) and (ii) for the next transform pass in the transform pass order, if a match exists; and

iv) identifying the current transform pass as a point of interruption, if a match does not exist.

15. The mobile handset of claim 14 wherein:  
the update package comprises a predetermined checksum for each memory bank in the subset of the plurality of memory banks for the interrupted transform pass.

16. The mobile handset of claim 14 wherein the update agent is capable of:  
initiating a recovery from the point of interruption in the transform pass order; and  
determining the first bank in the associated memory bank order for the interrupted transform pass for which a mismatch between a computed checksum and a predetermined checksum for a memory bank in the associated memory bank order for the interrupt transform pass occurs.

17. The mobile handset of claim 16 wherein the computed checksum and the predetermined checksum comprise a cyclic redundancy check (CRC) value.

18. The mobile handset of claim 16 wherein the computed checksum and the predetermined checksum comprise a MD5 hash value.

19. A method for recovering from interruption of a fault-tolerant process of updating a mobile handset comprising a plurality of memory banks from a first firmware version to a second firmware version, the update process comprising a plurality of transform passes and having a transform pass order, each of the plurality of transform passes performing a transform upon the plurality of memory banks in a memory bank order, the method comprising:

determining as a recovery transform pass, one of the plurality of transform passes interrupted during the update process, wherein each of the plurality of transform passes is associated with its own memory bank order, and wherein each transform pass updates a plurality of memory banks in the associated memory bank order for that transform pass;

determining as a recovery memory bank, one of the plurality of memory banks in the associated memory bank order for the interrupted transform pass during which update processing was interrupted; and

invoking the update process by performing an update of the recovery memory bank using the recovery transform pass.

20. The method according to claim 19 wherein determining a recovery transform comprises:

employing one of the plurality of memory banks as a decision maker bank for each of the plurality of transform passes, each of the decision maker banks identifying the last bank of the memory bank order for the corresponding one of the plurality of transform passes to be updated, the decision maker banks retrievable from an update package; and

determining an interrupted transform pass in the transform pass order, based upon the decision maker banks for the plurality of transform passes in the fault-tolerant update process.

21. The method according to claim 20 wherein determining an interrupted transform pass comprises:

- i) computing a checksum of a decision maker bank, beginning with the first transform pass in transform pass order;
- ii) comparing the computed checksum to a predetermined checksum retrieved from the update package to determine whether the computed checksum matches the predetermined checksum;
- iii) repeating (i) and (ii) for the decision maker bank of each of the subsequent transform passes in the transform pass order, if the computed and the predetermined checksums for a decision maker bank match; and
- iv) identifying as interrupted, a transform pass for which the computed and predetermined checksums for a decision maker bank do not match.

22. The method according to claim 21 wherein the computed checksum comprises one of a cyclic redundancy check (CRC) value and an MD5 hash value.

23. The method according to claim 20 wherein the update package comprises a set of executable instructions for converting a first firmware version to a second firmware version.

24. The method according to claim 19 wherein the method is employed during both an initial update attempt and during recovery from an interrupted update attempt.

Application Serial No. 10/790,340  
Appeal Brief  
October 5, 2009

**EVIDENCE APPENDIX**  
**(37 C.F.R. § 41.37(c)(1)(ix))**

- (1) United States Patent No. 7,082,549 ("Rao"), entered into record by Examiner in July 24, 2007 Office Action.

Application Serial No. 10/790,340  
Appeal Brief  
October 5, 2009

**RELATED PROCEEDINGS APPENDIX**

**(37 C.F.R. § 41.37(c)(1)(x))**

Not applicable.